

垃圾回收概述

一、常见术语

1.1 并行

两个或者多个事件在同一时刻发生，在现代计算机中通常指多个处理器上同时处理多个任务。

1.2 并发

两个或者多个事件在同一时间间隔发生，在现代计算机中一台处理器"同时"处理多个任务，这些任务只能交替运行，从处理器的角度看任务是只能串行，但从用户的角度看，这些任务是“并行”执行的，实际上是处理器根据策略不断切换知行合一这些“并行”的任务。

所以，并行和并发是从处理器的角度出发。但是在垃圾回收领域或者jvm领域，并行和并发在jvm中被重新定义了。

1.3 jvm中的并行

多个垃圾回收相关的线程在操作系统之上并发运行，这里的并行强调的是只有垃圾回收线程工作，java应用程序终止(或称为暂停)，因此类似ParNew工作的时候，一定发生了STW。

1.4 jvm中的并发

垃圾回收相关的线程并发运行（如果启动多个线程的话），同时这些线程会和java程序并发运行。

1.6 STW（stop-the-world）

停止一切，jvm中指停止一切java应用的线程。

1.7 安全点

jvm在执行一些操作的时候需要STW，但并不是任何线程在任何地方都能进入stw。

设置安全点的目的，当线程进入到安全点时，线程会主动停止。

1.8 Mutator

通常指我们的java应用线程。含义是可变的。这里的含义是因为线程运行，导致了内存变化。GC中通常需STW才能使Mutator暂停。

1.9 记忆集

Remember Set.简称Rset。记录不同代对象之间的引用关系。

1.10 Refine

在G1中主要指处理RSet的线程

1.11 Evacuation

转移、撤退、回收。

简称Evac。在G1中指的是发现活跃对象，并将对象复制到新地址的过程。

1.12 回收(Reclaim)

分区对象已经死亡或者已经完成Evac，分区可以被jvm再次使用

1.13 Closure 闭包

jvm中的一种辅助类，类似于iterator。通常提供了对内存的访问。_

1.14 GC ROOT

垃圾回收的根，jvm垃圾回收的过程中，需要从gcroot出发，标记活跃对象，确保正在使用的对象在垃圾回收之后都是存活的。

1.15 根集合（Root Set）

jvm的垃圾回收过程中，需要从不同的gc root出发，这些gc root有线程栈、monitor列表、jni对象等等。这些根就构成了根集合。

1.16 Full Gc

简称FGC,整个堆的垃圾回收动作。通常FGC是串行的，G1的FULL GC不仅有串行实现，在JDK10中还有并行实现。

1.17 再标记

Remark.指的是并发标记算法中，处理完并发标记之后，需要更新并发标记中Mutator变更的引用，这一步一般是需要STW的，不然标记过程是无法终止的。

二、回收算法概述